

DOI: 10.13382/j.jemi.2017.06.016

# 随机差分变异粒子群混合优化算法\*

曹文梁

(东莞职业技术学院 东莞 523808)

**摘要:**针对传统粒子群优化算法与差分进化算法都易出现早熟等问题,提出了一种随机差分变异粒子群混合优化算法。算法结合粒子群与差分算法的各自特点,首先采用差分变异方法产生试探性候选个体,再将其代入到粒子群速度更新公式,引导粒子飞行方向,从而扩大搜索空间,增强算法的全局勘探能力。为避免粒子陷入局部最优解,采用随机差分变异方式对当前最优粒子进行扰动,使算法在有效提高局部开采能力的同时,有效避免停滞现象的发生。算法分别在单峰及多峰等8个测试函数上与3个相关算法进行对比实验,实验结果表明,新的混合算法优于其他对比算法,有效提高了算法的性能。

**关键词:** 粒子群优化;差分进化;变异

**中图分类号:** TN911;TP181      **文献标识码:** A      **国家标准学科分类代码:** 520.604

## Hybrid algorithm based on particle swarm optimization with stochastic differential mutation

Cao Wenliang

(Dongguan Polytechnic College, Dongguan 523808, China)

**Abstract:** To solve the problem of premature convergence in traditional particle swarm optimization (PSO) and differential evolution (DE), a hybrid algorithm based on particle swarm optimization with stochastic differential mutation is proposed in this paper. Combining with the characteristics between PSO and DE, the new algorithm firstly generates a candidate individual using differential mutation, and then put the individual into velocity update formula leading flight direction of particle, which can expand the search space and enhance the global explorative ability of algorithm. Meanwhile, a stochastic differential mutation method is presented to disturb the current optimal particle in order to avoid the best particle being trapped into local optima, since which may cause search stagnation. The new algorithm compared with three related algorithms on 8 benchmark functions including unimodal and multimodal test functions. The experimental results show that the new hybrid algorithm outperforms other comparative algorithms and greatly improves performance of algorithm.

**Keywords:** particle swarm optimization; differential evolution; mutation

## 1 引言

函数优化问题一直是测试算法优化性能的经典问题,它具有广泛的应用背景,在商业、农业和工程等领域的各个分支几乎都存在这类问题。演化算法(evolutionary algorithm, EA)是一种源于自然进化的随机启发性智能算法,其算法特性对于解决函数优化问题具

有一定的天然优势,其中应用较为广泛的有差分进化算法(differential evolution, DE)<sup>[1]</sup>、粒子群优化算法(particle swarm optimization, PSO)<sup>[2]</sup>、遗传算法(genetic algorithm, GA)等。PSO算法是一种基于群体进化的随机搜索算法,由美国的Kennedy和Eberhart博士于1995年提出,其思想源于对鸟群及鱼群觅食行为的模拟。实验表明PSO具有较强的局部开采能力,收敛速度快,收敛精度高等特点,但同时存在易早熟及陷入局部最优的

收稿日期:2015-10      Received Date: 2015-10

\* 基金项目:东莞市社会科技发展项目(2013108101045)、东莞职业技术学院示范建设专项资金(政201614)资助项目

缺点。为此,科研人员不断尝试将 PSO 与其他算法相融合,以求改进 PSO 性能<sup>[3-10]</sup>。DE 算法由 Storn 和 Price 于 1997 年提出的另一种基于群体差异的智能进化随机搜索方法,包括变异、交叉、选择操作。与 PSO 相比,两者都具备原理简单、受控参数少且易于理解和实现等优点,但在针对某个具体问题最优解搜索的过程中,差分算法在前期具备较强的全局探索能力,但后期一般会出现收敛速度过慢或停滞以及收敛精度不高的不足。为充分发挥两者的优势,2001 年, Hendtlass 首次提出将 DE 与 PSO 相融合,至此,越来越多的科研工作者尝试将两者以不同的形式相融合,得到新的更为复杂但更加有效的优化算法<sup>[6-11]</sup>,并成功用于各种现实问题<sup>[4,5]</sup>。例如:Zhang 等人在 2003 年提出了一种混合差分粒子群优化算法 (DEPSO),该算法在奇数迭代中执行 PSO 操作,在偶数迭代中采用 DE 操作,并采用钟型曲线变异方式改善 DE 部分执行效果<sup>[6]</sup>。Das 等人<sup>[7]</sup>在提出的 PSO-DV 算法中,使用差分变异对 PSO 中速度更新公式进行扰动,并将 DE 中的选择策略引入到 PSO 中,有效提高算法性能。文献[8]中,种群被分成两个部分,分别在第一阶段的优化过程中采用 DE 和 PSO 进行优化,在第二阶段中,将 DE 中适应值较好的个体替换 PSO 群中数量相同的较差个体,从而帮助该部分群体个体逃离局部最优区域。文献[9]中,整体种群采用 PSO 算法进行优化,为增加种群多样性,对其速度更新公式中的认知部分采用 DE 算法进行深度进化。

为进一步改进上述混合算法性能,本文以一种新的融合方式将 DE 与 PSO 相结合,提出了一种新的随机差分变异粒子群混合优化算法 (hybrid algorithm based on particle swarm optimization with stochastic differential mutation, SDMPSO)。该算法将 DE 与 PSO 进行深度融合,首先采用差分变异方法产生一个试探性候选个体,再针对这个候选个体采用粒子群速度更新公式对原粒子飞行速度进行更新,引导粒子在下一时刻的飞行方向;同时,新算法在每次迭代中,通过随机差分变异方式对个体进行扰动,避免个体陷入局部最优。本文将新算法与 3 个相近算法在 8 个标准函数优化问题进行测试,实验结果表明,SDMPSO 有效平衡了算法的全局探索与开采能力,有效提高算法性能。

## 2 相关工作

### 2.1 标准 DE 算法

本质上 DE 算法是一种基于实数编码的具有保优思想的遗传算法,不同点在于 DE 利用父代不同个体之间的变异及贪婪选择策略产生子代。差分进化操作主要包括变异、交叉与选择操作,具体描述如下<sup>[1]</sup>。

1) 变异操作:假设种群中每个个体  $x_i$  的变异个体为  $v_i$  采用 DE/current-to-best/1 方式,则  $v_i$  表达式如式 (1)<sup>[10]</sup>。

$$v_i = x_i + F_1 \cdot (gbest_i - x_i) + F_2 \cdot (x_{r_1} - x_{r_2}) \quad (1)$$

式中:  $i, r_1, r_2, r_3 \in \{1, \dots, N\}$  为互不相同的索引, ( $N$  为种群规模),  $F_1, F_2$  为缩放因子。

2) 交叉操作:将上述变异后个体  $v_i$  与原个体按位进行交叉操作得到尝试向量。

$$u_{i,j} = \begin{cases} v_{i,j}, & rand \leq CR \text{ or } j = j_{rand} \\ x_{i,j}, & \text{其他} \end{cases} \quad (2)$$

式中:  $CR$  是交叉概率因子,  $j_{rand}$  是  $[1, D]$  之间的一个随机整数,从而保证  $u_i$  中至少有一维来自变异向量  $v_i$ , 避免与父个体向量  $x_i$  相同。

3) 选择操作:DE 采用贪婪原则选择  $u_i$  与  $x_i$  中适应值较优的个体进入下一次迭代中。

### 2.2 标准 PSO 算法

PSO 算法中的每个个体 (粒子) 通过在搜索空间内的移动来寻找最优解。每个粒子具有速度  $v_{i,j}$  与位置  $x_{i,j}$  两个属性,速度  $v_{i,j}$  依据个体最优位  $pbest_i$  和全局最优位  $gbest$  进行进化更新。其进化更新行为由如式(3)、(4)所描述<sup>[11]</sup>。

$$v_{i,j}(t+1) = \omega \cdot v_{i,j}(t) + c_1 rand_1 (pbest_{i,j} - x_{i,j}(t)) + c_2 rand_2 (gbest_j - x_{i,j}(t)) \quad (3)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad (4)$$

式中:  $i = 1, 2, \dots, N; j = 1, 2, \dots, D, N$  为种群规模,  $D$  粒子维数;  $\omega$  是惯性权值,  $\omega \in [0, 1]$ ;  $c_1$  和  $c_2$  为认知学习因子和社会学习因子,  $c_1, c_2 \in [0, 2]$ ;  $rand_1$  和  $rand_2$  是在  $(0, 1)$  区间上服从均匀分布的两个随机数。

## 3 SDMPSO 算法介绍

### 3.1 SDMPSO 算法

一系列研究表明,分析不同演化算法在优化问题中进化的特点,取长补短,相互融合是解决优化问题的一个有效方法,特别是在解决复杂的多峰优化问题中表现突出<sup>[5,13-14]</sup>。为此,本文提出了一种 SDMPSO 算法。该算法首先在解空间内随机产生  $N$  个个体,再依据 DE 变异方式 DE/current-to-best/1 依次生成一个试探性候选个体  $p_i$ , 并将它代入到 PSO 速度更新公式中,产生下一刻粒子飞行方向,对粒子  $x_i$  进行更新。同时,在每次迭代中,采用一个随机差分变异方式 (stochastic differential mutation, SDM) 对当前全局最优解进行扰动,从而帮助个体减小陷入局部最优位的可能性,并防止早熟现象的发生。

假设种群规模为  $N$ , 维度为  $D$ , SDMPSO 算法详细设计与操作步骤具体描述如下。

- 1) 在解空间内随机初始化  $N$  个个体。
- 2) 随机选择两个异于个体  $x_i$  的互不相同的两个个

$$p_{i,j} = \begin{cases} x_i + F_1 \cdot (gbest_j - x_{i,j}) + F_2 \cdot (x_{r1,j} - x_{r2,j}), \\ x_{i,j}, \end{cases}$$

式中:  $F_1$ 、 $F_2$  为缩放因子,  $rand$  是随机数,  $CR$  是交叉概率因子,  $j_{rand}$  是  $[1, D]$  之间的一个随机整数。

- 3) 将  $p_i$  代入到 PSO 速度更新公式中产生下一刻 ( $t+1$  时刻) 粒子飞行方向, 并依此更新粒子在  $t$  时刻的位置  $x_i(t)$ :

$$v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 rand_1(pbest_{i,j} - p_{i,j}) + c_2 rand_2(gbest_j - p_{i,j}) \quad (6)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad (7)$$

式中:  $\omega$  是惯性权值,  $rand_1$  和  $rand_2$  为  $(0, 1)$  之间的随机数。

- 4) 更新个体历史最优位  $pbest_i$  与当前全局最优位  $gbest$ 。

- 5) 对当前获得的  $gbest$  采用随机差分变异 (SDM) 策略进行变异操作:

$$gbest^* = gbest + rand_1 \cdot (x_{r1} - x_{r2}) \quad (8)$$

式中:  $rand_1$  为  $(0, 1)$  之间的随机数,  $x_{r1}$  和  $x_{r2}$  为群体中任意两个粒子。若  $gbest^*$  适应值  $f(gbest^*)$  优于  $gbest$  的适应值  $f(gbest)$ , 则用  $gbest^*$  置换  $gbest$ 。以最小优化问题为例, 即  $f(gbest^*) < f(gbest)$  则  $gbest = gbest^*$ 。

- 6) 当算法未达到结束条件时返回到步骤 2) 中。

### 3.2 SDM 策略

SDMPSO 算法中, 最优粒子  $gbest$  引导着群体向最优解前进(式(6)), 但若  $gbest$  陷入局部最优, 则易导致群体出现搜索停滞, 因此对其进行变异操作将有助于增加群体跳出局部最优的概率, Wang 等人<sup>[15]</sup> 提出一种柯西变异策略 (Cauchy mutation, CM), 对  $gbest$  的变异操作如下:

$$gbest^* = gbest + cauchy() \quad (9)$$

式中:  $cauchy()$  是服从标准柯西分布的随机数。该策略对部分测试函数效果良好, 但部分函数作用不大, 如 Schwefel 函数。

本文提出 SDM 策略对  $gbest$  进行扰动(式(8))。与 CM 策略不同, SDM 的变异大小及方向不再是遵循某种分布, 而是从群体中获得, 使  $gbest$  扰动更加有效。在后续实验中将对它做进一步分析。

### 3.3 算法计算时间复杂度分析

SDMPSO 算法主要包含 4 部分: 初始种群、候选个体生成、速度与位置更新、随机差分变异。易知, 初始种群和随机差分变异的时间复杂度都分别为  $O(N \cdot D)$  和  $O(1)$ ; 候选个体生成部分采用 DE 方式产生, 时间复杂度为  $O(N \cdot D)$ ; 速度与位置更新也在  $O(N \cdot D)$  内实现。综合上述分析, SDMPSO 的计算复杂度为  $O(N \cdot D)$ 。

体  $x_{r1}$  和  $x_{r2}$ , 采用 DE/current-to-best/1 变异方式生成候选个体  $p_i$  如式(5)。

$$rand \leq CR \text{ or } j = j_{rand} \quad (5)$$

其他

## 4 数值实验及分析

本文实验将被分成 3 个部分: 第 1 部分为算法性能测试, 将 SDMPSO 与 DE/current-to-best/1<sup>[10]</sup>、标准 PSO<sup>[2]</sup> 以及混合差分粒子群算法 DEPSO<sup>[5]</sup> 进行对比实验, 为了进行更加准确地比较, 以上算法参数设置与原文献保持一致; 种群规模  $N$  均取 40; 个体维度  $D$  取 30。各参与比较算法的具体参数设置如表 1 所示。第 2 部分分析随机差分变异策略 (SDM) 的有效性及其必要性。第 3 部分为参数敏感性分析, 观察 SDMPSO 中参数 CR 在取不同值时算法性能的变化情况。

表 1 算法参数设置

Table 1 Parameter setting of algorithms

	$c_1$	$c_2$	$w$	$CR$	$F_1$	$F_2$
PSO	1.5	1.5	0.5	-	-	-
DE	-	-	-	0.9	0.5	0.2
DEPSO	1.5	1.5	0.4	0.1	0.5	-
SDMPSO	1.5	1.5	0.4	0.9	0.5	0.2

### 4.1 实验设置

所有算法统一在 MATLAB2012 环境下通过 8 个经典测试函数进行实验。其中  $f_1 \sim f_5$  测试函数为单峰函数,  $f_6 \sim f_8$  为多峰测试函数。所有测试函数均在零点位取得全局最优值 0。具体测试函数如表 2 所示。

### 4.2 实验结果与分析

#### 4.2.1 算法对比分析

对 4 种算法进行的对比实验中, 所有算法在每个测试函数上分别运行 30 次, 每次均迭代 1 000 次, 记录全局最优值的平均结果如表 3 所示。每个测试函数中最优结果用粗体显示。为在统计意义上比较 4 个算法的性能, 采用 Friedman 检验法对结果进行显著性分析, 表 3 给出了 4 种算法的平均排名, 显然 SDMPSO 的秩(2.3)最小, 排名第一。

从表 3 可知, SDMPSO 在 4 种测试问题中都取得了最好的结果, 相比其它算法而言, SDMPSO 算法在解决多峰问题上表现较突出; 并通过表 3 中 Ranking 最后一行显示, SDMPSO 在 Friedman 检验中排名第一, 优于其他算法。限于篇幅的限制, 只选取  $f_3$  和  $f_7$  两个测试函数的收敛曲线如图 1 所示。特别指出, 在测试函数  $f_3$  中, 4 种算都陷入了局部最优值, 但 SDMPSO 表现较为稳定些。如何跳出局部最优是今后重点研究的问题。

表2 实验中使用的8个测试函数

Table 2 The 8 benchmark functions in experiment

	测试函数	查找位置	$f_{min}$	函数名
单峰	$f_1(x) = \sum_{i=1}^D x_i^2$	$[-100, 100]^D$	0	Sphere
	$f_2(x) = \sum_{i=1}^D ([x_i + 0.5])^2$	$[-100, 100]^D$	0	Step
	$f_3(x) = \sum_{i=1}^{D-1} [100 \cdot (x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$[-30, 30]^D$	0	Rosenbrock
	$f_4(x) = \sum_{i=1}^D (\sum_{j=1}^i x_j)^2$	$[-100, 100]^D$	0	Quadric
	$f_5(x) = \sum_{i=1}^D  x_i  + \prod_{i=1}^D  x_i $	$[-10, 10]^D$	0	Schwefel2.22
多峰	$f_6(x) = \sum_{i=1}^D [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12, 5.12]^D$	0	Rastrigin
	$f_7(x) = -20 \cdot \exp(0.2 \cdot \sqrt{\sum_{i=1}^D x_i^2} / D) - \exp((\sum_{i=1}^D \cos(2\pi x_i)) / D) + 20 + e$	$[-32, 32]^D$	0	Ackley
	$f_8(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]^D$	0	Griewank

表3 PSO、DE、DEPSO、SDMPSO 算法在6个测试函数上的结果均值

Table 3 The mean value of global optimum of PSO, DE, DEPSO and SDMPSO for 6 benchmark functions

函数	PSO	DE/current-to-best/1	DEPSO	SDMPSO
$f_1$	$1.49 \times 10^{-3}$	0.00	0.00	$1.53 \times 10^{-13}$
$f_2$	$6.66 \times 10^{-2}$	0.00	$5.02 \times 10^{-3}$	$3.60 \times 10^{-2}$
$f_3$	1.34	$2.26 \times 10$	$6.87 \times 10$	6.04
$f_4$	$5.68 \times 10^{-2}$	$3.78 \times 10^{-31}$	1.72	0.00
$f_5$	$-1.39 \times 10^{-3}$	$2.49 \times 10^{-143}$	$-6.98 \times 10^{-162}$	$-6.48 \times 10^{-32}$
$f_6$	2.03	$3.54 \times 10^{-2}$	$9.85 \times 10^{-3}$	0.00
$f_7$	$4.38 \times 10^{-2}$	$3.85 \times 10^{-1}$	1.16	0.00
$f_8$	$1.20 \times 10^{-3}$	0.00	$8.03 \times 10^{-3}$	$1.59 \times 10^{-3}$
Rankings	3.125	2.625	2.87	2.3

表4 不同变异策略算法比较结果

Table 4 Comparison results of SDMPSO with different mutation strategies

函数	SDMPSO-NULL		SDMPSO-CM		SDMPSO-SDM	
	SR/%	MNSM	SR/%	MNSM	SR/%	MNSM
$f_1$	100	-	100	$1.57 \times 10$	100	$2.34 \times 10^2$
$f_2$	46.30	-	80	2.01	100	6.60
$f_3$	57.61	-	76	1.33	100	6.04
$f_4$	100	-	0	0.00	100	$4.48 \times 10^2$
$f_5$	22	-	51.60	$4.72 \times 10^2$	69.31	$2.13 \times 10^2$
$f_6$	78.50	-	100	6.14	100	$1.37 \times 10$
$f_7$	89.07	-	100	2.94	100	$3.35 \times 10$
$f_8$	0	-	100	$1.67 \times 10$	97.71	1.15

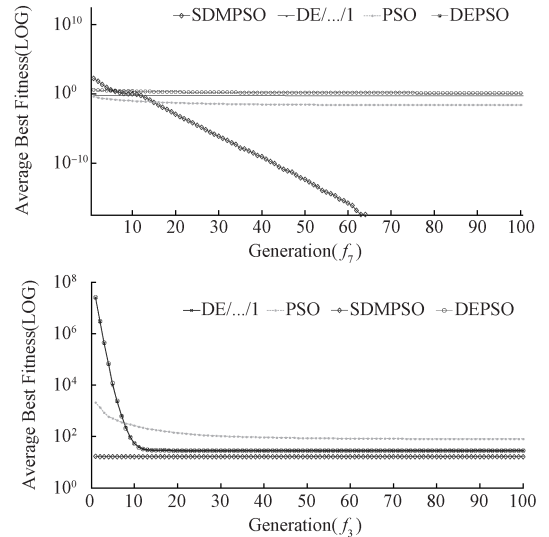


图1 PSO, DE/current-to-best/1, DEPSO 和 SDMPSO 在  $f_3$  和  $f_7$  上的收敛曲线

Fig. 1 Convergence curves of PSO, DE/current-to-best/1, DEPSO and SDMPSO for  $f_3$  and  $f_7$

4.2.2 变异策略分析

为研究 SDM 策略的有效性,将不含变异策略的 SDMPSO(SDMPSO-NULL)和集成 CM 策略的(SDMPSO-CM)作为对比算法与 SDMPSO(为表示明确,该实验部分记为 SDMPSO-SDM),实验结果如表4所示。实验记录了每个算法运行30次的变异策略成功率(successful rate, SR)(算法最终运行结果小于阈值  $1.0 \times 10^{-5}$  即视为成功

运行)以及 30 次的算法执行中变异成功次数的均值 (mean number of successful mutation, MNSM) (若变异后的  $gbest * 优于 gbest$ , 则认定变异策略成功)。

从表 4 可见,除  $f_8$  和  $f_6$  外,SDMPSO-SDM 在其它测试函数上 MNSM 均比 SDMPSO-CM 值高,表示 SDM 策略效果更好。同时可见,采用变异策略的 SDMPSO 算法均比没有采取变异策略的 SDMPSO-NULL 效果好,说明变异策略是非常有必要的,它在大多数情况下能增加个体逃离局部最优的可能性。就 SR 值表明 SDM 策略的鲁棒性和稳定性更好。

#### 4.2.3 参数敏感性分析

SDMPSO 算法中,CR 参数的设置大小对算法的性能往往有着很大的影响,CR 通过对 8 个测试函数分别在 (0, 1) 取 3 个不同的值,从算法 SDMPSO 收敛过程的观测可知,CR 取较大值 0.9 时性能达到最佳。由于篇幅的限制,如图 2 所示,以  $f_6$  为例,当 CR 取 3 个不同值时算法收敛到全局最优值的过程。

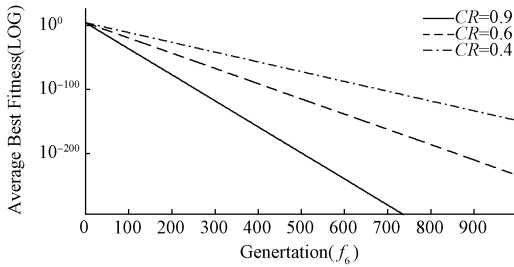


图 2 CR 在不同取值下 SDMPSO 全局收敛过程  
Fig. 2 Global convergence process of SDMPSO when CR gets different values

## 5 结 论

为进一步提高标准 PSO 算法与 DE 算法的性能,本文提出了 SDMPSO 算法。通过对 PSO 与 DE 算法特性分析,SDMPSO 算法采用 DE/current-to-best/1 变异方法产生一个试探性候选个体,从而扩大解的探索空间;该个体紧接着被代入到粒子群速度更新公式引导粒子在下一时刻的飞入方向;同时,新算法在每次迭代中,通过 SDM 策略对当前个体全局最优位  $gbest$  进行扰动,有效避免了个体陷入局部最优的可能性。

算法对 8 个单峰及多峰测试函数进行实验,实验结果显示,SDMPSO 算法性能上优于其它比较优化算法,能有效平衡对解空间的全局勘探及局部开采能力。同时,算法对 SDM 变异策略的必要性和有效性进行了进一步的分析,并给出相关重要参数的敏感性分析。值得注意的是算法在  $f_3$  中仍无法避免陷入局部最优。如何进一步提高优化算法,使其在更加广泛的测试函数中均取得最

优解是下一步需要考虑的问题。同时,在下一步的工作中,将新算法应用于盲信号分离等问题,进一步测试算法在解决实际问题中的有效性。

## 参考文献

[ 1 ] 刘波,王凌,金以慧. 差分进化算法研究进展[J]. 控制与决策, 2007, 22(7): 721-729.  
LIU B, WANG L, JIN Y H. Advances in differential evolution [J]. Control and Decision, 2007, 22 ( 7 ): 721-729.

[ 2 ] 康岚兰,董文永,田降森. 一种自适应柯西变异的反向学习粒子群优化算法 [J]. 计算机科学, 2015, 42(10):226-231.  
KANG L L, DONG W Y, TIAN J S. Opposition-based particle swarm optimization with adaptive Cauchy mutation [J]. Computer Science, 2015, 42 ( 10 ): 226-231.

[ 3 ] 汪济洲,鲁昌华,蒋薇薇. 一种新的基于混合粒子的粒化支持向量机算法[J]. 电子测量与仪器学报, 2015, 29(4): 591-597.  
WANG J H, LU CH H, JIANG WW. Novel granular support vector machine algorithm based on mixed granules [J]. Journal of Electronic Measurement and Instrumentation, 2015, 4, 29(4), pp: 591-597.

[ 4 ] 陈仁,朱霞,徐栋霞等. 基于改进型粒子群算法的卡箍直径检测算法研究. 仪器仪表学报, 2014, 35(8): 1837-1843.  
CHEN R, ZHU X, XU D X, a et al.. Spring clamp diameter detection algorithm based on the improved particle swarm optimization [J]. Chinese Journal of Scientific Instrument, 2014, 35(8): 1837-1843.

[ 5 ] MOORE P W, VENAYAGAMOORTHY G K. Evolving digital circuit using hybrid particle swarm optimization and differential evolution [J]. International Journal of Neural Systems, 2006, 16(3): 163-177.

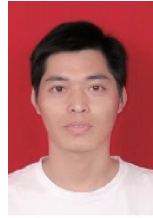
[ 6 ] RAMESH V, JAYABARATHI T, ASTHANA S, et al. Combined hybrid differential particle swarm optimization approach for economic dispatch problems [J]. Electric Power Components and Systems, 2010, 38 ( 5 ): 545-557.

[ 7 ] DAS S, KONAR A, CHAKRABORTY U K. Improving particle swarm optimization with differentially perturbed velocity [ C ]. Genetic and Evolutionary Computation Conference, 2005:177-184.

[ 8 ] 杨妍,陈如清,俞金寿. 差分进化粒子群混合优化算法的研究与应用 [J]. 计算机工程与应用, 2010, 46(25): 238-241  
YANG Y, CHEN R Q, YU J SH. Study on differential evolution-particle swarm optimization based hybrid

- optimization algorithm and its application[J]. *Computer Engineering and Applications*, 2010, 46(25): 238-241.
- [9] 郑灿赫, 孟广伟, 李锋, 等. 基于 PSO-DE 混合算法的结构可靠性优化设计[J]. *华南理工大学学报*, 2014, 42(9): 41-45
- ZHENG C H, MENG G W, LI F, et al. Structural reliability-based optimization design using PSO-DE hybrid algorithm[J]. *Journal of South China University of Technology*, 2014, 42(9): 41-45.
- [10] 汪慎文, 丁立新, 张文生, 等. 差分进化算法研究进展[J]. *武汉大学学报: 理学版*, 2014, 60(4): 283-292.
- WAN SH W, DING L X, ZHANG W SH, et al. Survey of differential evolution [J]. *Journal of Wuhan University: Natural Science*, 2014, 60(4): 283-292.
- [11] 栾丽君, 谭立静, 牛奔. 一种基于粒子群优化算法和差分进化算法的新型混合全局优化算法[J]. *信息与控制*, 2007, 36(6): 708-714.
- LUAN L J, TAN L J, NIU B. A novel ybrid global optimization algorithm based on particle swarm optimization and differential evolution [J]. *Information and Control*, 2007, 36(6): 708-714.
- [12] SHI Y, EBERHART R C. A modified particle swarm optimizer[C]. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2008)*, 2008: 69-73.
- [13] OMRAN M G H, ENGELBRECHT A P, SALMAN A. Bare bones differential evolution [J]. *European Journal of Operational Research*, 2009, 196(1): 128-139.
- [14] 陈义雄, 梁昔明, 黄亚飞, 等. 量子粒子群混合优化算法求解约束优化问题[J]. *小型微型计算机系统*, 2015, 36(2): 296-300.
- CHEN Y X, LIANG X M, HUANG Y F, et al. Hybrid algorithm based on quantum particle swarm optimization for saving constrained optimization problems[J]. *Journal of Chinese Computer Systems*, 2015, 36(2): 296-300.
- [15] WANG H, WU Z J, RAHNAMAYAN S, et al. Enhancing particle swarm optimization using generalized opposition-based learning [J]. *Information Sciences*, 2011, 181(20): 4699-4714.

### 作者简介



曹文梁, 2002 年于江西理工大学获得学士学位, 2008 年于江西理工大学获得硕士学位, 现为东莞职业技术学院副教授, 主要研究方向为智能计算、数据挖掘。  
E-mail: cw1222@tom.com

**Cao Wenliang** received B. Sc. and M. Sc. in computer science from Jiangxi University of Science and Technology in 2002 and 2008, respectively. Now, he is associate professor in Dongguan Polytechnic College. His main research interests include intelligent computing and data mining.